**Homework 2 – AWS CLI and Dynamo DB**

**Name:** Hiren Shah

**Date:** 9/10/2021 **School:** University of Maryland Global Campus

**Course:** SDEV 400\_6380

**Professor:** Errol Waithe

1. **(35 points)** Using Cloud9 AWS CLI environment, create a Table ‘Sensors’ with a Hash Key named ‘Sensor’. Additional Attributes in the database include:
   1. SensorDescription (String)
   2. ImageFile (String)
   3. SampleRate (Number)
   4. Locations (Set)
      1. Next use AWS-CLI to load 20 sensor from a stored JSON file. Make sure to add additional attributes on the fly, to at least 5 of the items
      2. Using AWS-CLI, write a statement that prints all items in the table
      3. Provide screen captures supporting the successful execution of each command

**Note:** I created the DynamoDB table named Sensors with Hash Key named Sensor of Type string. I created this in the AWS Cloud9 terminal using the following commands:

**aws dynamodb create-table \**

**--table-name Sensors \**

**--attribute-definitions \**

**AttributeName=Sensor,AttributeType=S \**

**--key-schema AttributeName=Sensor,KeyType=HASH \**

**--provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5**

Text

Description automatically generated

Figure . Part A. Showing DynamoDB table named Sensors with Hash Key named Sensor of Type string. This was created in the AWS Cloud9 terminal. Shah, Hiren. 2021. ©

Text

Description automatically generated

Figure . Part B. Showing the rest of the output when entering the AWS CLI in terminal. This was created in the AWS Cloud9 terminal. Shah, Hiren. 2021. ©

Graphical user interface, application

Description automatically generated

Figure . Showing the empty Sensors table in the AWS DynamoDB cloud environment. Shah, Hiren. 2021. ©

I loaded the items into the Sensors table by running a command which loads records from the JSON file. This command loads the items in a whole batch so the that it can all be done with a single command.

**aws dynamodb batch-write-item --request-items file://JSON\_SENSOR\_SHAH\_HW2.json \**

**--return-consumed-capacity INDEXES --return-item-collection-metrics SIZE**

Text

Description automatically generated

Figure 4. Showing Output of batch righting from the JSON file of 20 elements with arbitrary attributes. Shah, Hiren. 2021. ©

The next screenshot shows the 20 items loaded onto DynanoDB Sensors table.

Graphical user interface

Description automatically generated with medium confidence

Figure 5. Showing the batch write of items loaded into DynamoDB table. Shah, Hiren. 2021. ©

Next, I ran the following command to return all items in the Sensors table..

***aws dynamodb scan --table-name Sensors***

Text

Description automatically generated

Figure 6. Showing partial output of returning all items on the JSON file from the cloud9 terminal. Shah, Hiren. 2021. ©

1. **(40 points)** Using the AWS SDK and Python within Cloud9, create a DynamoDB table and functionality as described:
   1. Table named Courses to hold attribute including Subject (e.g. SDEV), CatalogNbr (e.g. 400), Title (e.g. Secure Programming in the Cloud), NumCredits (e.g. 3), and a CourseID (e.g. 001). The CourseID should represent the Hash Key for the Courses table.
   2. Input 10 Course items of your choice. Note that all attributes are required for each item entered in the Courses table.
   3. Provide a command line simple interface that allows a user to search for a title given the Subject and CatalogNbr. The search program should continue to loop until the user requests to exit. Also, if both Subject and CatalogNbr are not entered, the program should request the data be re-entered.

Text

Description automatically generated

Figure . Showing partial code from HW\_Q3Database.py with table “Courses” and key type: Hash. Shah, Hiren. 2021.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case** | **Input** | **Expected Output** | **Actual Output** | **Figure Reference** |
| 1 | SDEV, 460, N | The course is Software Security Testing | The title of SDEV is 460 Software Security Testing.  Next the program exits | 8 |
| 2 | CMIS, 141, y, CMIS, 242, Y, hello, worlf, n | It will display the course title for CMIS 141.  It will display the course title for CMIS 242.  Then it will say course not found on third try. Then it will exit. | The title of CMIS is 141 Introductory Programming.  The title of CMIS is 242 Intermediate Programming.  Course not found!  Process exited with code: 0 | 9 |
| 3 | SDEV, 495, sure, SDEV, 460, n | It will display the course title for SDEV 495 and SDEV 460. The it will ask user to search again, and because I enter ‘n’, it will exit program | The title of SDEV is 495 Current Trends & Projects in Comp-Sci.  The title of SDEV is 460 Software Security Testing.  Process exited with code: 0 | 10 |

Text

Description automatically generated

Figure . Showing the actual output of test case 1. Shah, Hiren. 2021. ©

Text

Description automatically generated

Figure . Showing the output of test case 2. Shah, Hiren 2021. ©

Text

Description automatically generated

Figure . Showing the output for test case 3. Shah, Hiren 2021. ©

1. **(15 points)** Use the AWS CLI to delete all DynamoDB tables that were created for this exercise. Provide AWS CLI responses and the commands used to successfully complete this task in your report.

The following command deletes both tables per assignment question # 3.

***aws dynamodb delete-table --table-name Sensors***

***aws dynamodb delete-table --table-name Courses***

Text

Description automatically generated

Figure . Showing the output of deleting table Sensors. Shah, Hiren. 2021. ©

Text

Description automatically generated

Figure . Showing the output of deleting table Courses. Shah, Hiren. 2021. ©

Graphical user interface, text, application, email

Description automatically generated

Figure . Showing DynamoDB environment and all tables deleted. Shah, Hiren. 2021. ©

**References**

Wilinski, R. (2020). *This cheat sheet covers the most important DynamoDB CLI query examples and table manipulation commands*. Dynobase. <https://dynobase.dev/dynamodb-cli-query-examples/>